The restaurant business in the US is huge. It comprises about 4% of the US GDP, where 99% of restaurants are family-owned. This industry currently employs~ 15 million people, or about 10% of the working population, and is projected to continue to grow by double digits over the next ten years. However, failure rate of restaurants is 60% within the first year and 80% failure rate with 5 years. The most common reason for failure is due to suboptimal location. My project will build a model that predicts restaurant success at census-tract or zipcode-levels in the US based on demographics and popularity of current restaurant categories (food type, and venue type, and ethnicity). This model will be of value to potential restauranteurs and banks to inform whether a planned business will be successful in a chosen location. Further, the model can inform what restaurant category would be most successful as well as find optimal locations for proposed restaurant with a city.

The demographic information will come from the US Census Bureau and the restaurant popularity index will generated from a yelp review dataset that includes restaurant reviews from 7 metroplexes. NLP techniques be used to 1) create food category features by vectorization and 2) generate popularity index label in order to train the model. I will evaluate multiple supervised models (logistic regression, random forest, gradient boost and support vector machines), as well as deep learning models with tensorflow keras packages. Unsupervised learning can be used on vectorized language data to uncover review subtypes in order to help construct popularity labels. I will also use geopandas to overlay restaurant review data by census tract boundaries in order to generate census tract-level evaluation.

The first challenges are to deal with data size. There are millions of reviews and feature generation with NLP can increase the column number on the order of 100s. Dimension reduction with UMAP and PCA may simplify vectorized features, but Dask may be necessary to speed up the deep learning tasks with parallelization. Spark would be nice to try, but may have a bit of a learning curve to handle some of the more complex feature engineering steps due my unfamiliarity with this platform. Other major hurdles will be in engineering the popularity label. Ideally it would be nice to have a clean success rate label, but that data would take a very long to time to achieve, as restaurant open and closure rates are not easy to find at this scale. The star rating is not always the best way to evaluate the success of a business as it may be biased, inaccurate or even ridiculous. Also, the number of positive and negative reviews overall may be unbalanced. Therefore, upsampling or downsampling may be used to overcome these issues where necessary.